1. Describe the requirements of the problem with a simple document that lists the rules of the database in the problem domain language. Then from that list of rules and notes highlight the list of possible nouns and actions you identified. I'm expecting this to be a short 1 or 2 pages document.

**Problem Domain:**

The SHMS efficiently coordinates the logistics of healthcare management. It integrates patient care, staff scheduling, equipment tracking, and financial processes, providing an all-encompassing solution for healthcare facility management.

**Database Rules:**

1. Each entity (Patient, Doctor, etc.) is assigned a unique identifier.
2. Patients can have multiple appointments, with each appointment associated with one patient, one doctor, and possibly one piece of medical equipment.
3. Treatment plans are personalized for patients and are overseen by doctors.
4. Medical equipment is tracked and allocated per department.
5. Billing information is connected to each patient and reflects their treatment and equipment usage.
6. Administrative staff manage and facilitate various non-medical patient-related activities.
7. Departments serve as organizational units within the healthcare facility.

**Identified Nouns (Potential Entities):**

1. Patient
2. Doctor
3. Appointment
4. Treatment Plan
5. Medical Equipment
6. Department
7. Admin Staff
8. Billing Information

**Identified Actions (Operations):**

1. Schedule Appointment
2. Reschedule Appointment
3. Cancel Appointment
4. Update Treatment Plan
5. Record Treatment
6. Allocate Medical Equipment
7. Update Medical Equipment Status
8. Manage Billing Information
9. Update Patient Records
10. Manage Department Information
11. Assign Admin Staff Duties
12. Analyze the problem and create a conceptual model in UML using a tool of your choice (e.g., LucidChart, Enterprise Architect, ArgoUML, Visual Paradigm, ERwin, TOAD) as discussed during class and provided in the references and resources below. Additional requirements and clarifications will be provided in the #general channel on Slack. The diagram must contain at least three classes, at least one to many relationship and one many to many. All relationships, except generalization, must have full multiplicity constraints and labeled as appropriate. Classes must have proper names, descriptions, and attributes with domain types. Key attributes and derived attributes must be marked. Don't build a model with more than 10 entities.

A diagram of a software company

Description automatically generated with medium confidence

1. From the Conceptual Model, construct a logical data model expressed as an ERD using a language of your choice (other than UML) and a tool of your choice. The logical data model may not have any many-to-many relationships, so introduce association entities as needed.

A diagram of a computer program

Description automatically generated with medium confidence

1. From the logical model, define a relational schema in at least BCNF. Using functional dependencies, show that the schema in in at least BCNF.

**Schema with Relationships** (underscore for Primary Key, Italic/bold for Foreign Key)

* Patient(patientID, name, dateOfBirth, address, ***insuranceID***)
* Doctor(doctorID, name, primaryCare, schedule)
* Appointment(appointmentID, ***patientID***, ***doctorID***, date, time)
* TreatmentPlan(treatmentPlanID, ***patientID***, ***doctorID***, diagnosis, treatment)
* MedicalEquipment(equipmentID, name, availability)
* Department(departmentID, location)
* AdminStaff(staffID, name, role)
* BillingInformation(billingID, ***patientID***, amountDue, paymentStatus)
* Manage(***patientID***, ***billingID***)
* Assigned(***patientID***, ***treatmentPlanID***)
* Schedule(***patientID***, ***appointmentID***)
* Associatedwith(***appointmentID***, ***treatmentPlanID***)
* Take(***doctorID***, ***appointmentID***)
* Provide(***doctorID***, ***treatmentPlanID***)
* Own(***departmentID***, ***equipmentID***)

1. Create a set of SQL data definition statements for the above model and realize that schema in SQLite3 by executing the script from the SQLite3, the console or Node. You can use DB Browser to generate these statements. Show that the tables were created and conform to the constraints through screen shots or other means.

A screenshot of a computer

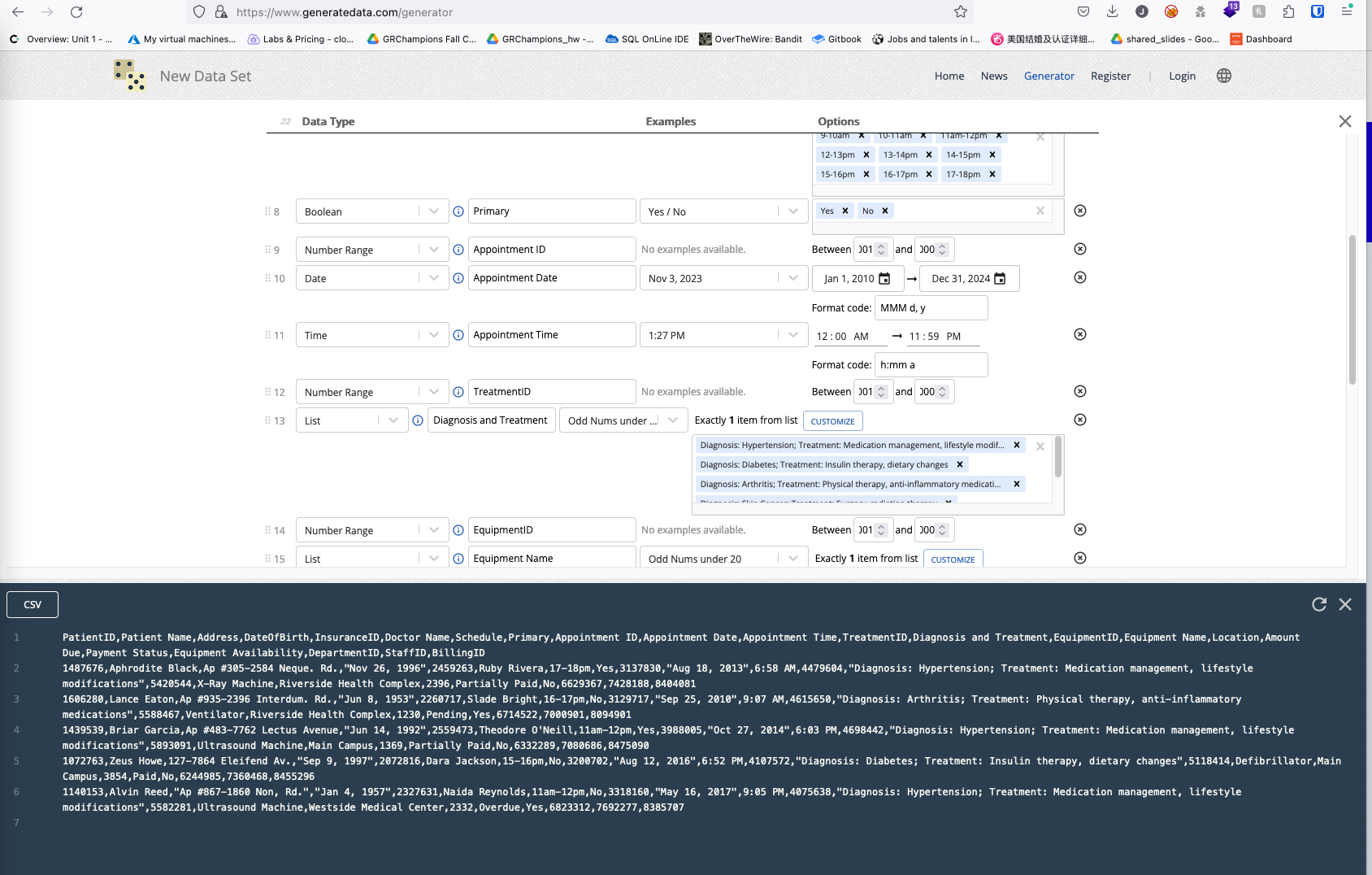
Description automatically generated

A screenshot of a computer

Description automatically generated

1. Populate the tables with test data.

Generated 500 rows of data.



1. Define and execute at least five queries that show your database. At least one query must contain a join of at least three tables, one must contain a subquery, one must be a group by with a having clause, and one must contain a complex search criterion (more than one expression with logical connectors). Experiment with advanced query mechanisms such as RCTE, PARTITION BY, or SELECT CASE/WHEN.

A screenshot of a computer

Description automatically generated